

```
import seaborn as sns

import matplotlib.pyplot as plt

sns.set(style="white",color_codes=True)

import pandas as pd

data=pd.read_csv("/games.csv")

data.head()
```



	id	rated	created_at	last_move_at	turns	victory_status	winner	in
0	TZJHLIjE	False	1.504210e+12	1.504210e+12	13	outoftime	white	
1	l1NXvwaE	True	1.504130e+12	1.504130e+12	16	resign	black	

```
data["turns"].value_counts()
```

```
53    303
45    302
51    299
57    297
39    297
```

```
...
```

```
216     1
208     1
176     1
218     1
201     1
```

```
Name: turns, Length: 211, dtype: int64
```

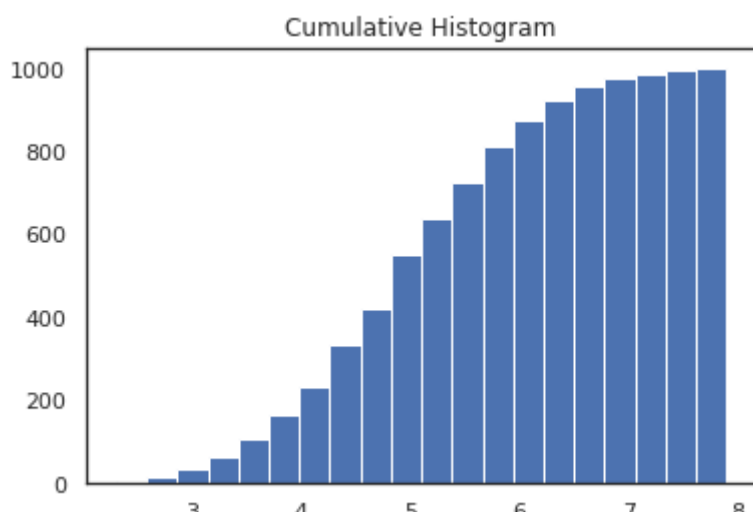
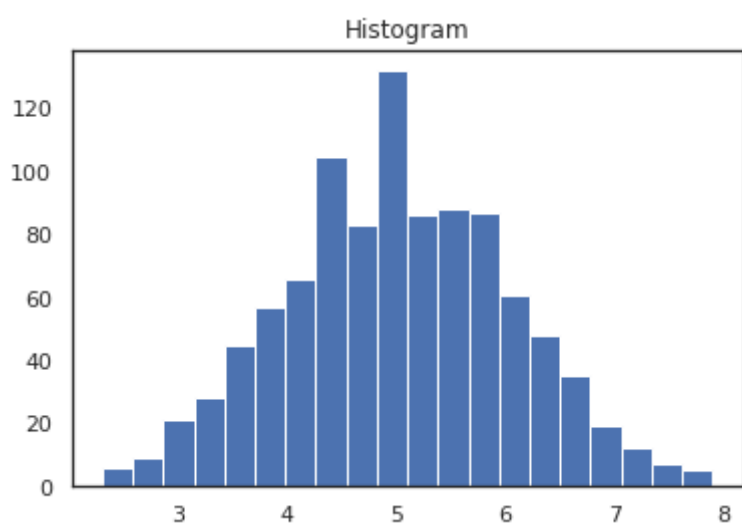
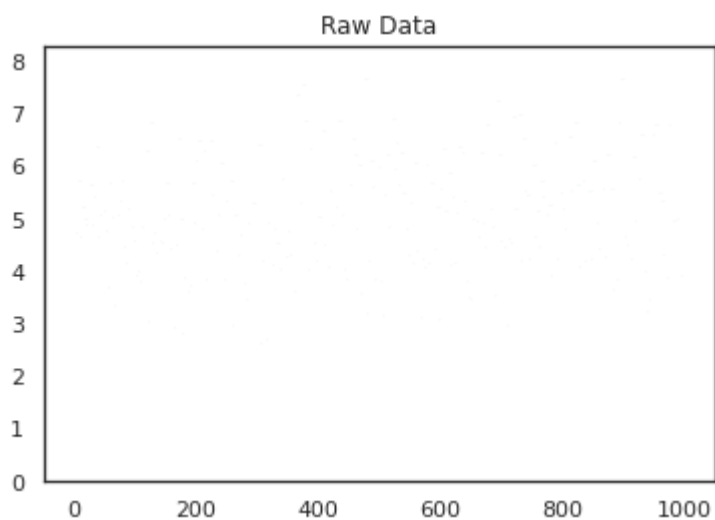
```
import matplotlib.pyplot as plt
import numpy as np
```

```
# Use numpy to generate a bunch of random data in a bell curve around 5.
n = 5 + np.random.randn(1000)
```

```
m = [m for m in range(len(n))]
plt.bar(m, n)
plt.title("Raw Data")
plt.show()
```

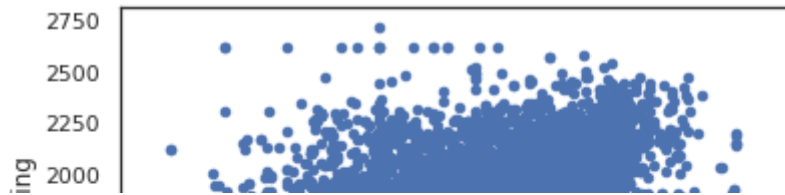
```
plt.hist(n, bins=20)
plt.title("Histogram")
plt.show()
```

```
plt.hist(n, cumulative=True, bins=20)
plt.title("Cumulative Histogram")
plt.show()
```



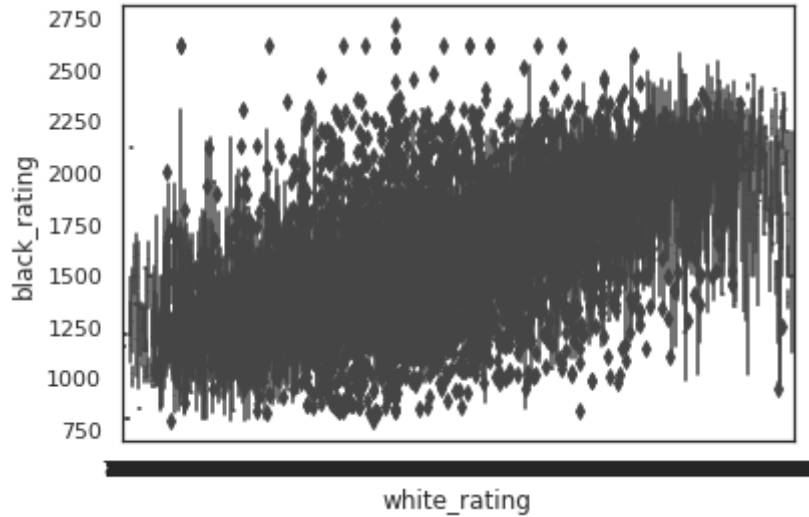
```
data.plot(kind="scatter",x="white_rating",y="black_rating")
```

```
*c* argument looks like a single numeric RGB or RGBA sequence, which should be avoided
<matplotlib.axes._subplots.AxesSubplot at 0x7fc83913a5d0>
```

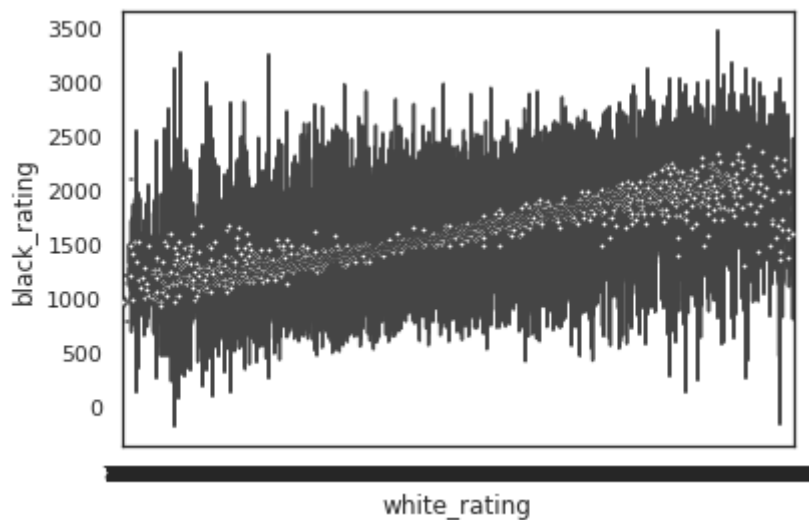


```
sns.boxplot(x="white_rating",y="black_rating",data=data)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fc8387f5c50>
```



```
sns.violinplot(x="white_rating",y="black_rating",data=data,size=6)
plt.show()
```

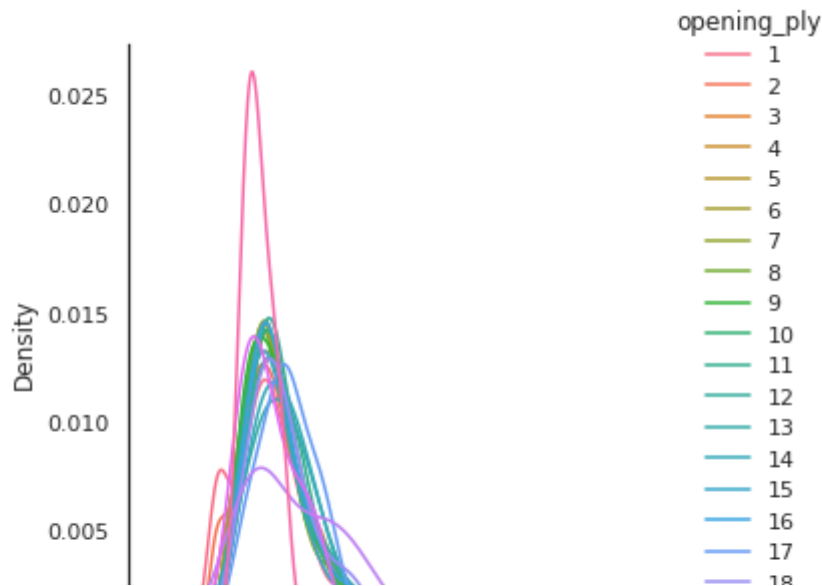


```
sns.FacetGrid(data,hue="opening_ply",size=5).map(sns.kdeplot,"turns").add_legend()
plt.show()
```

```

/usr/local/lib/python3.7/dist-packages/seaborn/axisgrid.py:337: UserWarning: The `size`
warnings.warn(msg, UserWarning)
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:316: UserWarning: Dat
warnings.warn(msg, UserWarning)
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:316: UserWarning: Dat
warnings.warn(msg, UserWarning)

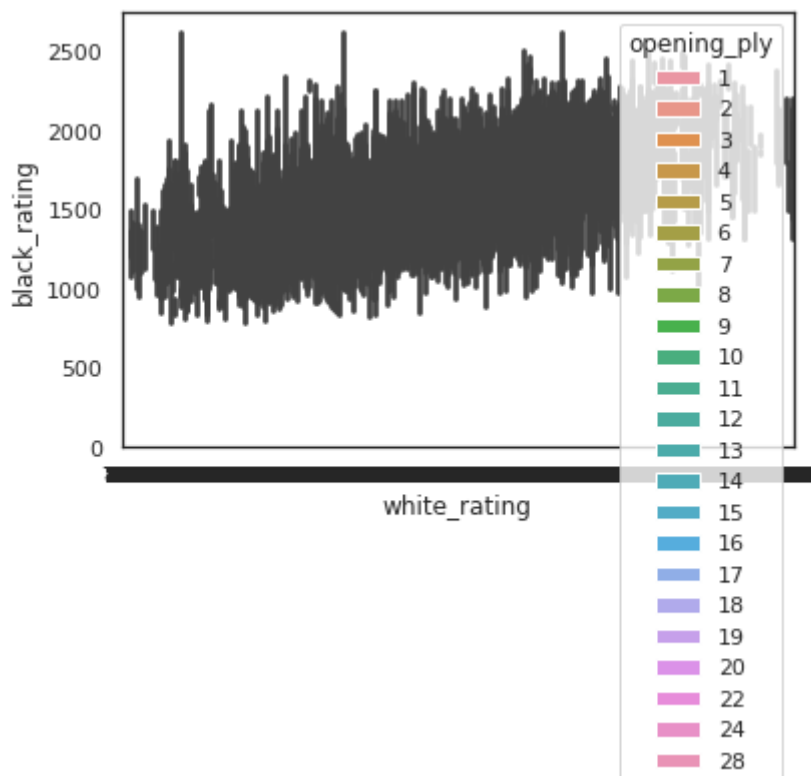
```



```

sns.barplot(x="white_rating",y="black_rating",data=data,hue="opening_ply")
plt.show()

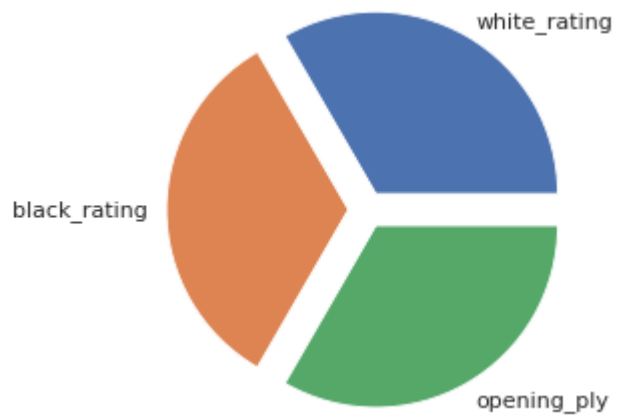
```



```

labels = ["white_rating","black_rating","opening_ply"]
sizes = [50,50,50]
plt.pie(sizes,labels=labels,explode=(0.1,0.1,0.1))
plt.axis("equal")
plt.show()

```



```
sns.displot(data["white_rating"],bins=25,kde=True)
```

```
<seaborn.axisgrid.FacetGrid at 0x7fc809899990>
```

